

Endrekursion mit Scheme

Endrekursion
mit
Scheme

[Racket]

Endrekursion mit Scheme

Muster einer endrekursiven Funktion

```
(define
  (endrekursive-Fkt Parameter Akkumulator)
  (if
    (Abbruch? Parameter)
    Akkumulator
    (endrekursive-Fkt
      (update Parameter)
      (Bearbeitung Parameter Akkumulator)))
  )
)
```

Endrekursion mit Scheme

Muster einer endrekursiven Funktion

```
(define
  (endrekursive-Fkt Parameter Akkumulator)
  (if
    (Abbruch? Parameter)
    Akkumulator
    (endrekursive-Fkt
      (update Parameter)
      (Bearbeitung Parameter Akkumulator)))
  )
)
```

einer oder mehrere Parameter

Endrekursion mit Scheme

Muster einer endrekursiven Funktion

```
(define
  (endrekursive-Fkt Parameter Akkumulator)
  (if
    (Abbruch? Parameter)
    Akkumulator
    (endrekursive-Fkt
      (update Parameter)
      (Bearbeitung Parameter Akkumulator)))
  )
)
```

*Wir benötigen einen
Zwischenspeicher*

Endrekursion mit Scheme

Muster einer endrekursiven Funktion

```
(define
  (endrekursive-Fkt Parameter Akkumulator)
  (if
    (Abbruch? Parameter)
    Akkumulator
    (endrekursive-Fkt
      (update Parameter)
      (Bearbeitung Parameter Akkumulator)))
  )
)
```

Verzweigung ...

... mit Abbruchbedingung

Endrekursion mit Scheme

Muster einer endrekursiven Funktion

```
(define
  (endrekursive-Fkt Parameter Akkumulator)
  (if
    (Abbruch? Parameter)
    Akkumulator
    (endrekursive-Fkt
      (update Parameter)
      (Bearbeitung Parameter Akkumulator)))
  )
)
```

*Ergebnis steht im
Akkumulator*

Endrekursion mit Scheme

Muster einer endrekursiven Funktion

```
(define
  (endrekursive-Fkt Parameter Akkumulator)
  (if
    (Abbruch? Parameter)
    Akkumulator
    (endrekursive-Fkt
     (update Parameter)
     (Bearbeitung Parameter Akkumulator)))
  )
)
```



Endrekursion mit Scheme

Muster einer endrekursiven Funktion

```
(define
  (endrekursive-Fkt Parameter Akkumulator)
  (if
    (Abbruch? Parameter)
    Akkumulator
    (endrekursive-Fkt
      (update Parameter) bearbeitete Parameter ...
      (Bearbeitung Parameter Akkumulator)))
  )
)
```

Endrekursion mit Scheme

Muster einer endrekursiven Funktion

```
(define
  (endrekursive-Fkt Parameter Akkumulator)
  (if
    (Abbruch? Parameter)
    Akkumulator
    (endrekursive-Fkt
      (update Parameter)
      (Bearbeitung Parameter Akkumulator)))
  )
)
```

... und das aktuelle Zwischenergebnis

Endrekursion mit Scheme

Beispiel einer endrekursiven Funktion

```
(define
  (endrekursive-Fkt Parameter Akkumulator)
  (if
    (Abbruch? Parameter)
    Akkumulator
    (endrekursive-Fkt
      (update Parameter)
      (Bearbeitung Parameter Akkumulator)))
  )
)
```

Endrekursion mit Scheme

Beispiel einer endrekursiven Funktion

```
(define  
  (Abbruch? Liste)  
  (null? Liste))
```

```
(define  
  (update Liste)  
  (rest Liste))
```

```
(define  
  (Bearbeitung Liste Akkumulator)  
  (cons (first Liste) Akkumulator))
```